REMARKS

Reconsideration of the above-identified application in view of the following analysis and remarks is respectfully requested.

Claims 1, 2, and 4 – 11 are pending in this case.

§112 Rejection

In response to the Examiner's §112 rejection of dependent claim 3, the Applicants have elected to cancel claim 3.

§103 Rejections

Claims 1, 2, and 4 – 11 have been rejected under 35 USC §103 as being unpatentable over Tso [US 6,088,803] in view of Hall [US 2004/0054928]. The Applicants respectfully traverse the rejection, as detailed below.

The Office Action notes that Tso "is not executable with code that extracts the object", but asserts that "it would have been obvious for one of ordinary skill in the art to secure the object of Tso by wrapping it in the wrapper of Hall" (Office Action, page 3).

The Applicants respectfully traverse the above rejection on the grounds that, because of technical reasons which would be readily apparent to a person ordinarily-skilled in the art, it is not possible to combine Tso and Hall in such a fashion to produce the result of the present invention, as explained in detail, following.

The present invention is directed to a method for preventing the activation of a malicious object passing through a checkpoint in a path from a source to a destination, such as an e-mail message in transit to a receiver on a self-contained client device. Tso is likewise directed to a similar goal, involving virus checking during download

from a server to the client device over a network. Both the present invention and Tso seek to prevent the activation on the client device of a malicious object (e.g., a virus) that is in the process of being sent to the self-contained client device, by preventing the malicious object from being fully installed on the client device.

In contrast, however, Hall is directed to a method for detecting intrusions within a device or network, as Hall's title and abstract clearly indicate. The nature of Hall's method, environment, and context is therefore entirely different from that of Tso, as is elaborated herein.

In particular, Hall seeks to prevent the activation of an existing, fully-installed executable object within the device or network environment by an unauthorized individual. The executable object whose activation Hall seeks to prevent is not a malicious object, but rather a normal object that is loaded in the environment (Hall calls this "latent software"). Hall gives the example of a Linux operating system "*ls*" command [Hall, paragraph 0045] as being such an object. The activation of such an object does not by itself pose any threat — Hall merely seeks to prevent their activation by personnel who are not authorized to use them, and to signal a notification of intrusion in case of an attempt by unauthorized personnel. To facilitate this goal, Hall discloses a "wrapper script" that has the same name as the object (and thus looks the same to an intruder) and replaces the object in the object's original location. The object itself is moved to a new "non-standard" location that is not known to the intruder. The intruder thus believes the wrapper script to be the object which is to be activated and activates the wrapper script. The wrapper script contacts an external monitoring server which determines whether or not the user is authorized to activate the object. If the monitoring server confirms the user, the wrapper script simply executes the original object now residing in the new "non-standard" location.

Otherwise, the wrapper script aborts the intruder's attempt and the monitoring server signals an intrusion alert.

Because of these fundamental differences, there is no way that Hall can be used in combination with Tso to accomplish the desired goal of the present invention — of preventing the activation of a malicious object being downloaded over a network or being sent by e-mail. Specific reasons precluding such a combination are readily apparent to a person ordinarily-skilled in the art, and include, but are not limited to, the following enumerated points:

1. Hall is specifically configured to prevent the activation of normal software which is not malicious in nature ("latent software"). Tso, however, is specifically configured to prevent the activation of malicious software that has no normal function. This difference alone indicates an incompatibility that precludes any reasonable expectation of success in combining the two technologies.
2. In order to use Hall in preventing the activation of an object, that object has to be loaded (or "fully installed") in the environment of a target server on the network. [Hall paragraph 0007] *"...The method comprising the steps of loading monitored latent software on the target server and monitoring. Attempts to execute monitored latent software on the target server from the client are received and it is determined whether the attempt to execute the monitored latent software by the client is authorized prior to completely executing the monitored latent software."* Thus, Hall cannot be used in combination with Tso because Tso prevents the object from being loaded on the destination device. This also precludes any reasonable expectation of success in combining the two technologies.

3. As noted above, for Hall to prevent the activation of a malicious object (e.g., a computer virus) that malicious object has to be loaded on the target server. Deliberately loading a malicious object, such as a computer virus, on a server is precisely what Tso is configured to prevent. There is therefore no reasonable expectation of success in combining Hall with Tso.

4. Tso has a goal of preventing the activation of a malicious object wherever and however the malicious object might be activated. In contrast, Hall can prevent the activation of an object only when the object is invoked by an unauthorized user external to the device on which the object is installed (e.g., "Attacker Client" 32 external to latent software on "Target Server" 22 in Figures 1 and 3). Hall is thus useless in preventing the activation of a malicious object that might be triggered internally within the destination device which receives the malicious object, which is a goal of Tso. Again, there is no reasonable expectation of success in combining Hall with Tso.

In addition:

5. Hall's "wrapper script" does not comprise the object whose activation is to be prevented, as is provided by the present claims. Hall's wrapper script is a completely separate file that is located in a completely separate location from that object. Thus, Hall (as well as Hall in combination with Tso, which does not provide for an envelope file having executable code) fails to meet the limitations of the present claims (e.g., claim 1) which provide for creating an envelope file comprising both executable code and the object.

6. Hall's "wrapper script" moreover does not result in the extraction of the object from an envelope file, as provided by the present claims. Hall's wrapper script is restricted to executing an object that requires no extraction prior to

execution. Thus, Hall (as well as Hall in combination with Tso) fails to meet the limitations of the present claims (e.g., claim 1) which provide for code for extracting the object from an envelope file.

As detailed above in points **1.** through **4.**, it would be clear to a person of ordinary skill in the art that combining Tso with Hall fails to have a reasonable expectation of success, which is a requirement for a *prima facie* case of obviousness, as mandated *inter alia* by MPEP 2143.02.

In addition, as detailed above in points **5.** and **6.**, even if Hall were combined with Tso, the combination fails to meet all the limitations of the present claims, which is also a requirement for a *prima facie* case of obviousness, as mandated *inter alia* by MPEP 2143.03.

Conclusion

In view of the above analysis and remarks it is respectfully submitted that claims 1, 2, and 4 – 11 are indeed in accordance with 35 USC §103 and 35 USC §112, and are in condition for allowance. Accordingly, a notice of allowance is respectfully and earnestly solicited.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: June 6, 2007